Readability project:

Number of lines according to my calculations = 995

Abstract:

This project is focused on determining the level of texts and if necessary simplifying these texts. These levels are based on the requirements for the European levels (A1-A2-B1-B2-C1-C2). An added feature is the simplifications of text with a level higher than B1 to that B1 level (intermediary level).

General: Checklist

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Task | Lines | Description |
| All | **Logo** | O | Let’s all think about a logo (name) for our project. We need something easy, that turns up in a search machine (relatable) with a nice sound.  My suggestion: EasyText, it is easily found in a search engine, sounds “fun” and it not that hard to make a logo with only the ET or the entire title |
| X | **Abstract** | 0 | A few short lines describing our project |
| Helke + ev Machteld | **Training set (100)** | 0 | **We need a training set**. So it is pressing we decide if we cover all fields or just one field (or a few) like literature/essays/economical, scientifical,…. Papers / ….  **The set exists of sets (**how do we make this library EXEL? I can ask Mike): “level, text” |
| All | **Importance** | 0 | We have to decide how important each part of the level indicator is going to be. I suggest tenses and words will have the most weight. But what percentage do we assign to each part? |
| Kimberly | **Corpus** | 0 | I am still looking for a corpus, but still haven’t found one yet. Therefore, I propose that if I don’t find one today (24/12) we go to plan B (I make one myself based on the texts we have).  In that case I will not help on the trainings set itself since this will take a lot of work(however, to do this I need as many text as possible as soon as possible) |
| Kimberly | **Interface** | 100?? | Okay I am going to take the interface for my part. It is supposed a fun and easily usable interface without any confusions. |
| All | **Test group (optional)** | 0 | This step is only necessary if we actually get our code working/running. We need a varied test group for our program (*just for fun: if it is actually accurate enough, if the interface works well for them, ….)*  *🡪 I know some teachers (high school and (maybe) teaching modules for English), my sister and my mom that represent 3 parts of our goal group*  *🡪Helke said that she had some victims as well?* |

Part 1: Checklist

Hint: write all in function structure, we can paste them in a separate document as well which makes it easier to keep our file neat (**at least this is what I understood from class, is this correct?)**

**Grammar link** : <http://www.examenglish.com/CEFR/cefr_grammar.htm>

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Task | Lines | Description |
| X | **Language control** | 15 lines | A user doesn’t always enter the correct text. Build in a control to check if the language is actually English.  Tip: check for function words like (an,a,the) in a text.  (*if we decide to branch out ;) 🡪 this gets necessary to decide which language you have to run (ge. Dutch, Spanish, German, French, …)* |
| X | **length of lines**  **length of words**  **word count**  **sentence count** | 20 lines | An important part of level appointment is text structure. The average length of sentences and words can be an indicator in this |
| Kimberly | **Word level** | 150 lines | 1. Spelling checker build in (optional, aka. When I find out how to work this) = connected to the lemma abstraction 2. Lemma abstraction (see spelling corrector)  * Tokenization * Plural, suffixes, verb forms, named entities, bye bye punctuation, grammatical words (linking words), delete numbers, abbr.’s (+1 hard).  1. Connecting the word to the correct level  * Outsiders (words not in the program are automatically counted C2)  1. Connecting the text to a word level  * *To understand a text people need to know at least 90% of a text, therefore I will make the program starting from C2 going downward (les work for the program)* |
| X | **Tense determination** | 150 lines | Each level (hint, look at the link above)  The program has to be able to recognize the tense and the level that that tense belongs to.  I suppose we can use the same structure as for words . Start from the top by looking to the most difficult structures and then go down.  !!!! take the irregular verbs into account as well!!! I suggest that this person makes an exception list of the irregular verbs with their tenses in each category (this can also be used for the words part but this way this person has more lines. |
| X | **The level of co-reference** | 100 lines | ? |
| X | **??** |  | **Here stopped my knowledge of the first part, Helke I think you have that page?** |
| Kimberly | **Level calculation** | 100 lines | The different parts combined in a level calculator |

Part 2: Checklist

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Task | Lines | Description |
| Kimberly | Sentences level | 50 lines | * Split long sentences on linking words * Make passive sentences active |
| X | Tenses | 50 lines | Convert tenses from higher than B1 to B1 level |
| Helke | Word level | 200 lines | * **Machine learning** 🡪 teaching the program to decide on the change that a word has to be replaced or deleted * Applied to the program (actually changing/deleting words) * Ej. Deletion of unnecessary adjectives * Avoid abbr. (regex) 🡪 write them in full * **Optional**: make sure that some words are underlined, when pressing them some optional synonyms should occur |
| X | Syntax | 50 lines | Simplify the co-reference chain |